Website URL: <localhost:3000>
GitHub final version (Dec 7): <https://github.com/hazraimran/cmpt415-gamification-python/tree/master>

Qingrui (Rachel) Li

- Accomplished task
  - Use Monaco Editor together with judge0 to compile student code in the background.
    - This Editor can run just like a normal python compiler out there. The black part is where the code is written, and the "Customer Input" is a special place for the input function. The "Output Window" is the place to show the result of the compile. When you finish your code, just press “Run” Button.
    - See Figure 2
  - Create the grading system
    - It can grade everything except drawing graph like input, function, if-else, for loop and so on.
    - Grade system will run when press the Run button
    - Give students scores by comparing the students’ codes with the answer code (When grade the “function” code, just add the test code on firebase and it will be read and use the grade the code. If you need to change the test code, you can do so directly in the firebase backend in questions.questionCode)
    - See Figure 2, 3and 5
  - Design the UI
    - See Figure 1 and 2
  - Create the Hint system
    - If students do not try to write the code, they will need some point to see the hint. (The points will reduce when pressing hint button) If they try, they can free to get the answer.
    - The Hint for each question is going to be a separate pop-up window
    - See Figure 2 and 4
  - Read and store data from firebase:
    - Read questions database (include answers code and questions) from firebase
    - Read, store and update students’ score in firebase and web
  - Store students code when they write in editor (Can see in firebase to double check students answer)
    - The answer to each question written by each student and whether it is completed or not will also be stored in
  - Write the rules about how to use the code challenges system in firebase in RulesCodeProblem.docx in project folder

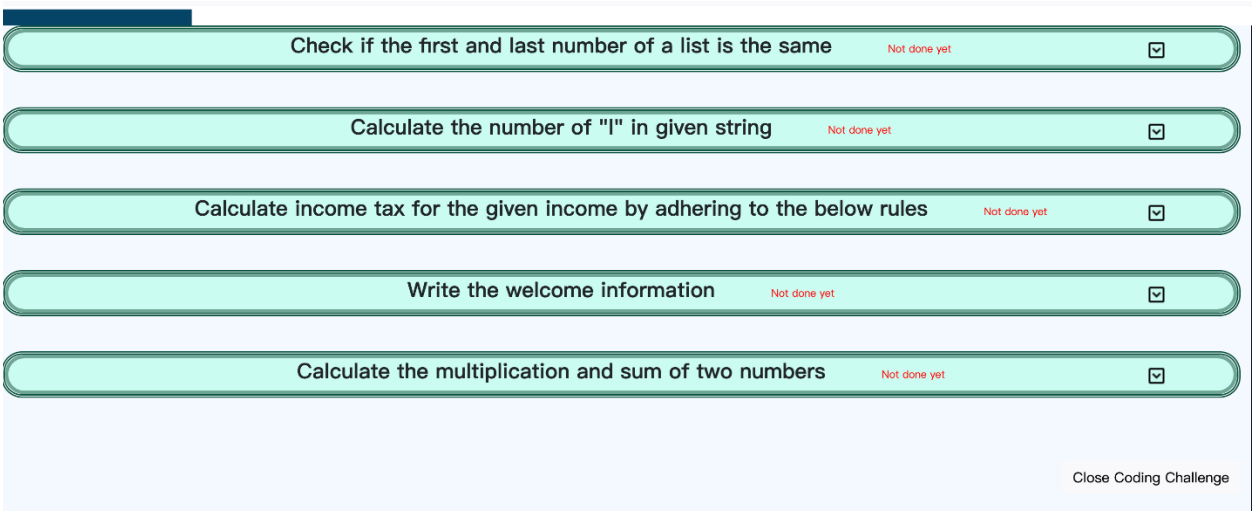

Figure 1

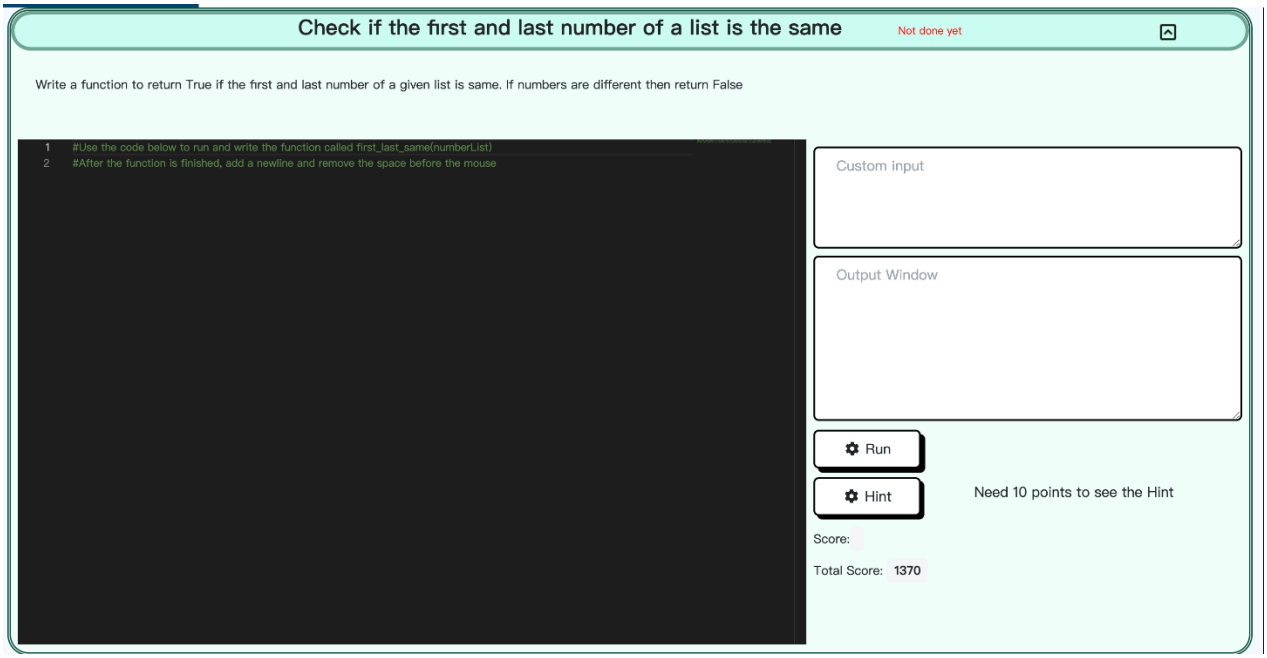

Figure 2

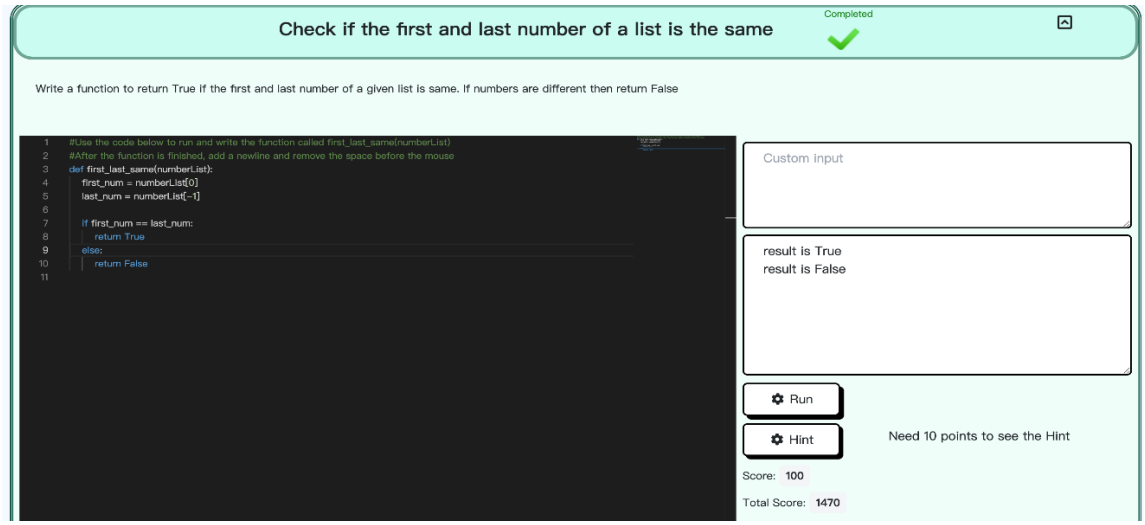

Figure 3

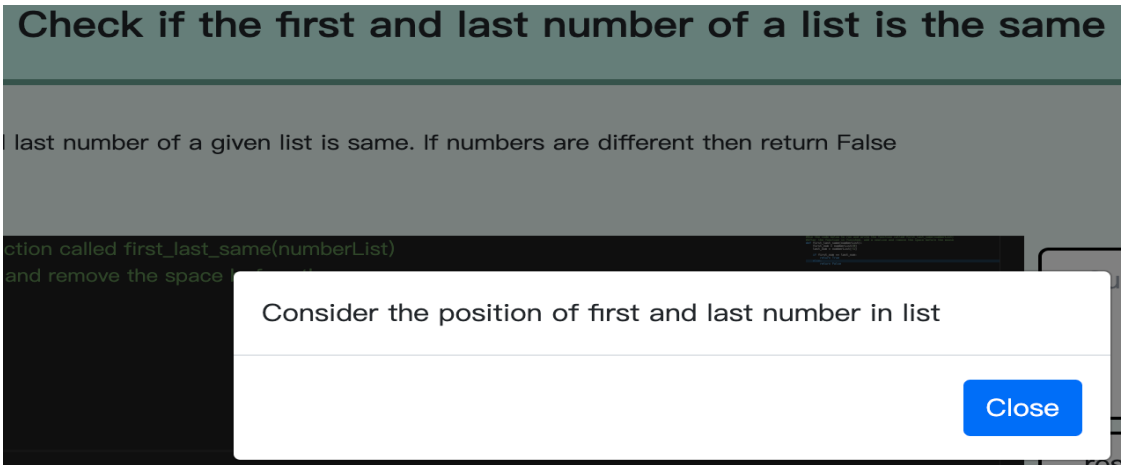

Figure 4: Hint system

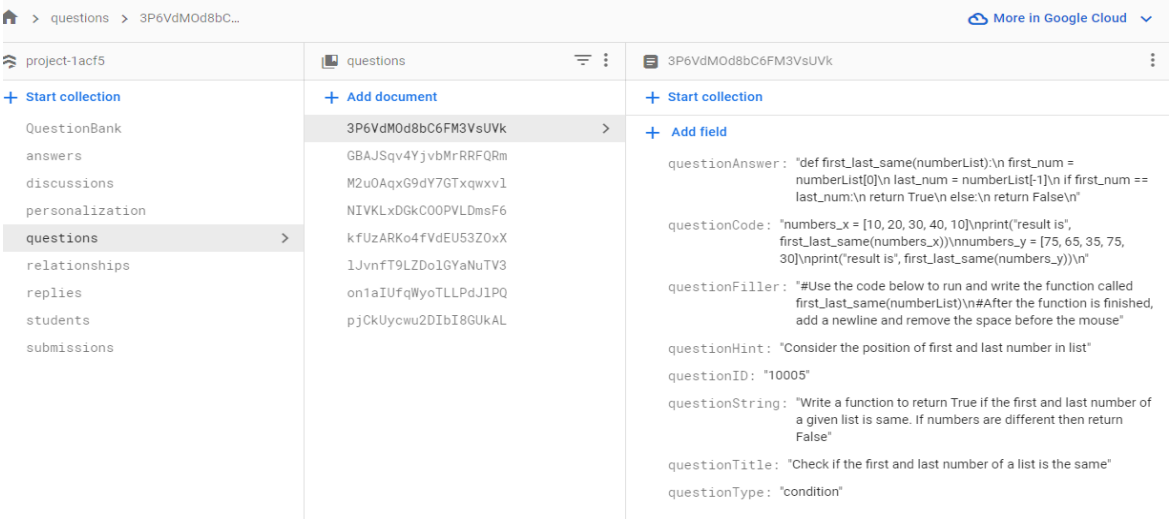

Figure 5: questions database

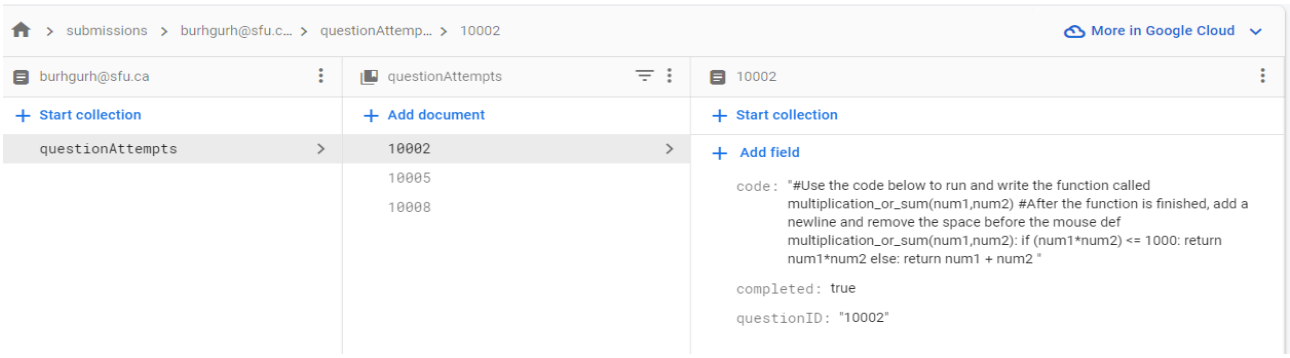

Figure 6: submissions database

Field Name:	Description
questionAnswer:	• Use \n to connect between each line of code • Unless something like loop or function is used, no space between codes • Each line of code in a loop or function, for example, needs to have two spaces in it. If it's something like a loop with a loop in it, you need to add 4, and so on. (for, def and other similar first sentences without spaces)
questionCode:	• If students just need to write the function, add the test code here to run their code. • Each code line needs to be followed by an "\n" • If they do not need to write function, just use "#" instead.
questionFiller:	Notes on questions, such as function names, and how to use
questionID:	Unique ID to distinguish the question
questionString:	Details of Question
questionTitle:	Title of question
questionHint:	Hints
questionType:	• Type of different questions • Like condition_statement can called"condition"

Figure 7: Part of screenshot in RulesCodeProblem.docx

- Future work
    - Setup for CodeChallenges part
        - Setup firebase and set up to two databases in firebase: questions and submissions (if not create database called students, also need to setup it)
        - Setup RapidAPI judge0 to get your own API(header called 'X-RapidAPI-Host') https://rapidapi.com/judge0-official/api/judge0-ce
        - Create the .env file from .sample-env in folder
            - NEXT_PUBLIC_FIREBASE_API_KEY=
            - NEXT_PUBLIC_FIREBASE_AUTH_DOMAIN=
            - NEXT_PUBLIC_FIREBASE_DATABASE_URL=
            - NEXT_PUBLIC_FIREBASE_PROJECT_ID=
            - NEXT_PUBLIC_FIREBASE_STORAGE_BUCKET=
            - NEXT_PUBLIC_FIREBASE_MESSAGING_SENDER_ID=
            - NEXT_PUBLIC_FIREBASE_APP_ID=
            - NEXT_PUBLIC_FIREBASE_MEASUREMENT_ID=
            - NEXT_PUBLIC_RAPIDAPI_KEY=
            - Last one is to put your own rapid api get before. Others are to put the information get from firebase (you can go to project setting to get all the information)
        - Need to add questions in firebase database questions. More details about how to setup and store questions is in the RulesCodeProblem.docx (It also has some examples can just copy and paste in database.) See Figure 5 before.
    - Understanding Code
        - Landing.js (Page in Figure 1) (ModuleComponent.js connect it to code challenge part) (Both of them in Component folder)
            - Read the student data and questions data from firebase and call the class component from CodeProblem.js to show on the web
        - CodeProblem.js (Page in Figure 2) (In editor folder in Component folder)
            - Includes all the feature in figure2 and deal with the information get from Landing.js
            - Submit function – do the part connected to judge0 and compiled the code, then compare the students' code with correct answer and provided it as
        - questions.js (In data folder)
            - Get the question information from firebase database
        - Students.js (In data folder)
            - Get student information, add scores, delete scores in firebase.
- Document shared on discord
    - Seenan and I found many links related to the code challenges. We share privately in the discord and discussed them together. Then, Seenan did the summary and shared them in discord group.
    - Some text Editors that can be implemented for users to type code client-side:
        - Monaco (Works well with React): https://www.npmjs.com/package/@monaco-editor/react
        - Ace (Written in JS) https://ace.c9.io/

        - CodeMirror(Can customize features more(Restrict to one-line/custom auto-fill)): https://codemirror.net/5/mode/python/
    - Execution of the code :
        - Can use judge0 API https://judge0.com/
    - Can execute in the server with NodeJS:
        - Executing Python with child processes in NodeJS: https://medium.com/swlh/run-python-script-from-node-js-and-send-data-to-browser-15677fcf199f https://javascript.plainenglish.io/how-to-run-python-script-using-node-js-6b351169e916
        - Using PythonShell module in nodeJS: https://www.geeksforgeeks.org/run-python-script-using-pythonshell-from-node-js/
        - API to create sandbox in nodeJS to run Python: https://www.npmjs.com/package/docker-python-sandbox